# Getting Started With *DSM* Grandchild Board

## Using VHDL

Prof. Dr. Martin J. W. Schubert, Electronics Laboratory,

OTH Regensburg, Regensburg, Germany

# Getting Started With DSM Grandchild Board

## Using VHDL

**Abstract.** This communication presents the DSM conversion grandchild board, which is a daughter board to the *ADA* board, which is a daughter board to different *DE1-SoC* board or other *DEx* boards from *Terasic* with compatible user header, e.g. *DE2, DE2-70, DE2-115*.

# 1 Introduction

## 1.1 Objectives and Organization of this Document

This document illustrates how the A/D Converter (ADC) and D/A Converters (DAC) on the *ADA* board can be assembled to form a delta-sigma (ΔΣ) modulator (DSM) operating as ADC. This dommunication is focused to the analog part of the ΔΣ ADC. The significantly more complex digital part is given by the author.

This document assumes that you are familiar with the *DE1-SoC* board or a similar *DEx* board and the *ADA* daughterboard that carries the *DSM* board. The required introductions *Getting Started with DE1-SoC Board* and *Getting Started with ADA Daughter Board* are available from the author's homepage [15].

**The organization** of this document is as follows:

Section **1** is this **introduction**,
Section **2** makes the user familiar with the ***DSM*** board **hardware**,
Section **3** introduces first tests with the ***DSM*** board using **software**,
Section **4** assembles a ΔΣ A/D/A conversion system with *ADA* and *DSM* boards,
Section **5** draws relevant **conclusion** and
Section **6** offers **references.**

## 1.2 Tools

### 1.2.1 *DE1-SoC* Hardware

This document assumes that you are familiar with the *Terasic's* [2] *DE1-SoC* board using an *Intel Cyclone V FPGA* [2] or a similar board *DEx* board with the same general-purpose input/output (GPIO) user header. The version of your *DE1-SoC* board can be identified at [3]. *DE1-SoC* board revisions *F* and *G* differ in a printed company label only. To get it from the internet, go to [4] to find and download *DE1-SoC_v.5.1.2_HWrevF_SystemCD.zip* [5] or a later version and download it. It contains amongst other things important documents such as *DE1-SoC User Manual* [6] and *Schematic* [7]. On the computer system of OTH Regensburg you will also find the CD on drive K:\Sb\ [8]. Do not use any other manual revision to follow this documentation. The differences are sometimes considerable.

### 1.2.2 *Quartus II* [9] and *ModelSim* [10] Software Tools

It is assumed that you have *Intel's ModelSim [10] and Quartus II 13 [9]* software available. To download this freeware for your private PC you have to sign in at *Intel* [11]. At OTH Regensburg's PC pools of faculties *EI* and *IM* this software is installed. At faculty *EI* also *Quartus II 8* is installed supporting the older *DE2* boards with *Cyclone II* FPGAs, because they are no longer supported for *Quartus II* versions greater than 13.1.

### 1.2.3 Use of *VHDL*

The *IEEE standard VHDL Language reference manual* [12] is difficult to read. *Qualis VHDL Quick Reference Card* [13] and *1164 Packages Quick Reference Card* [14] are compact but difficult to understand. Feel free to find your own sources.

VHDL is not case sensitive. In the following, **`KEYWORDS`** will be written in **`ALL CAPITAL LETTERS`** and user defined names in **`lowercase letters`**. Exception: Capitalized initials are used for composed self-made names, e.g. *AddressBus* or *DataBus*. Self-made data types begin with *t_*, e.g. *t_StateVector*.

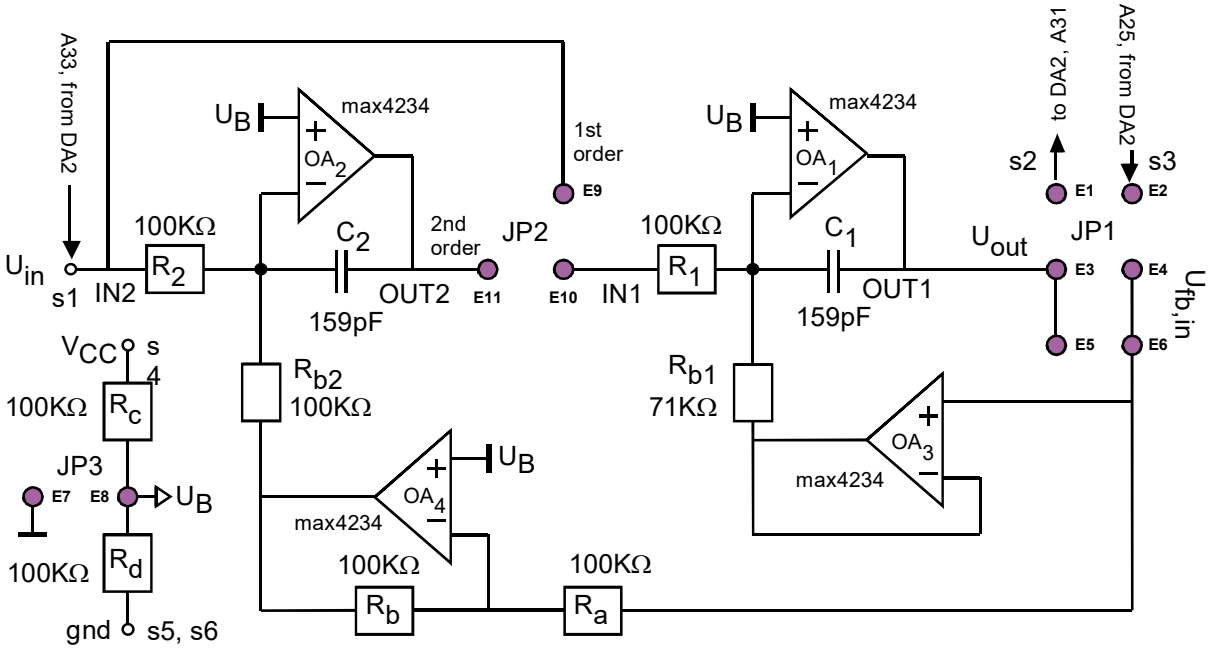
## 1.3 Acknowledgements

The author would like to thank *Terasic Technologies* [1] for admission to use screen copies of *Terasic* documentation for teaching purposes in this lectures.

```
At 19.09.2014 08:49, Terasic - Dong Liu wrote:
  Dear Martin,
  Thank you for using DE boards to teach VHDL. Yes, you can open all
  DE design resources for teaching purpose. Thank you!
  Best Regards,
  Doreen Liu
```

# 2 The DSM Grandchild-Board Hardware

**(a) DS2-Board Schematics: analog system or ΔΣ modulator, configurable of 1st and 2nd order**



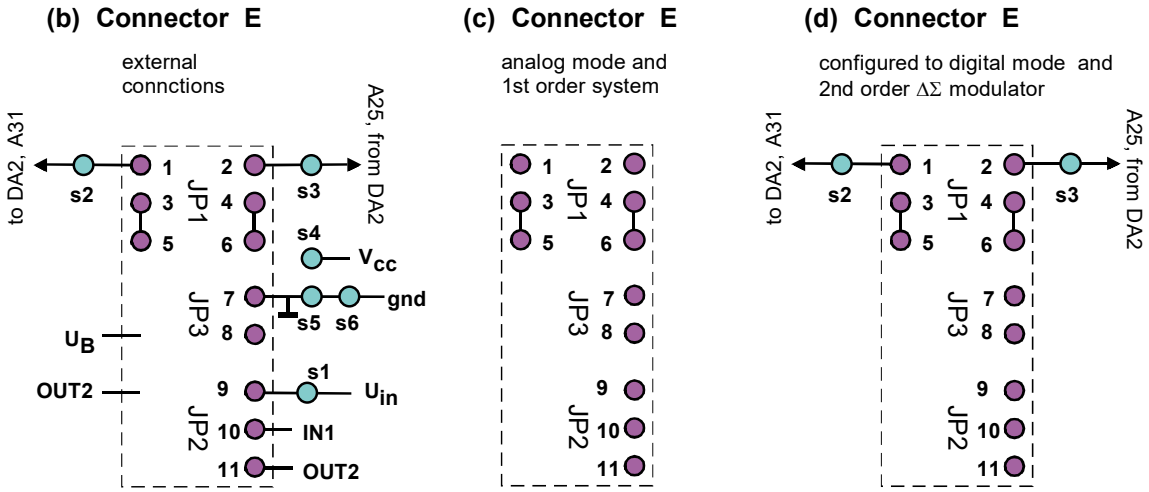




**Fig. 2.1:** (1) DSM-Board Schematics, cutoff frequency is $f_g$=10KHz.

Draw in Fig. 2.1(c) a jumper to set a 1st order system. Pins: ,
a jumper to connect on the *DSM* board's $U_{out} \rightarrow U_{fb,in}$ : .
Draw in Fig. 2.1(d) a jumper to set a 2nd order system. Pins: ,
a jumper to connect *DE1-SoC* board's $U_{out}$ → ADA/A31: ,
a jumper to connect *DE1-SoC* board's $U_{fb,in}$ ← ADA/A25: .

The *DSM* grandchild board is plugged under the *ADA* daughter board in the plug-line nearer to the *ADA* board. This automatically connects the wires drawn green in Fig. 3.2. The more distant plug-line is intended for disconnected parking of the *DSM* board.

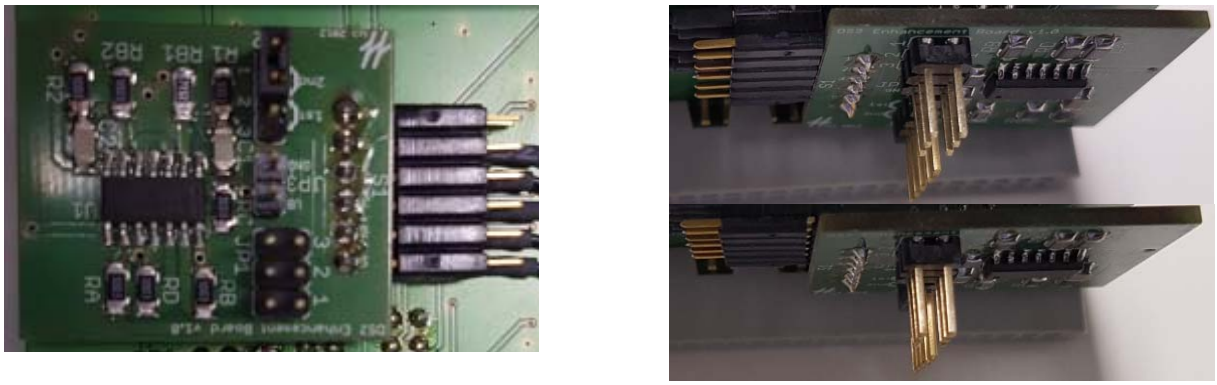
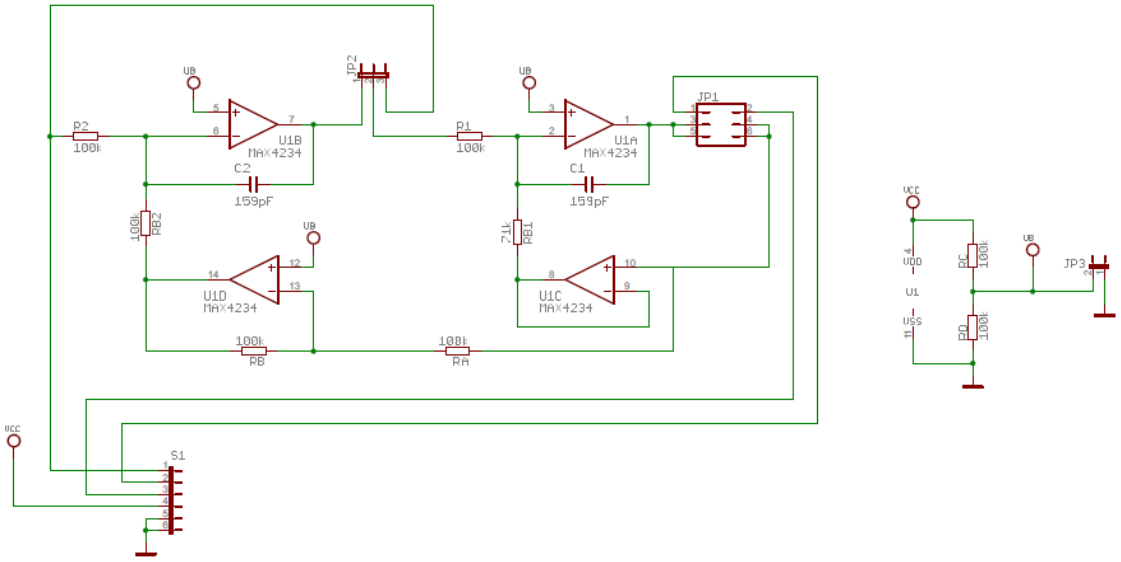
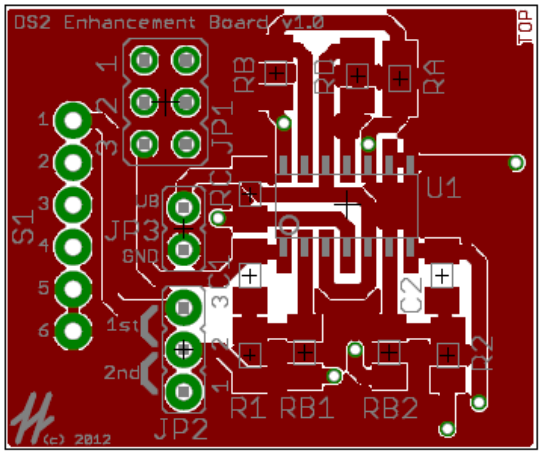


**Fig. 2.2: (a)** *DSM* board under *ADA* board, **(b)** electrically connected and **(c)** parking position

(a)




(b)




(c)

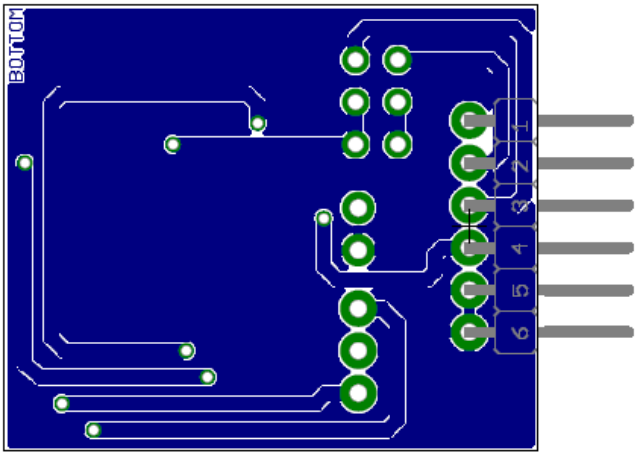



**Fig. 2.3:** Eagle design of the *DSM* boad: (a) Schematic, (b) top and (c) bottom view of the layout

# 3 Testing the *DSM* Analog Board

## 3.1 Testing the *DSM* Board in a Stand-Alone Mode

We will now operate the *DSM* board as stand-alone 1st or 2nd order analog system with lowpass characteristics. Therefore: → Set jumper E3-E4 on *DSM* board.

The *DSM* board gets its input signal from pin A33 which is wired to E10 of the *DSM* board. This connection is sketched as green wire *W1* in Fig. 3.2. Wires *W2* and *W3* remain disconnected in this subsection by not setting jumpers E1-E3 and E2-E4.

A 1st order system is obtained by setting a jumper connecting E10-E9;
A 2nd order system is obtained by setting a jumper connecting E10-E11.

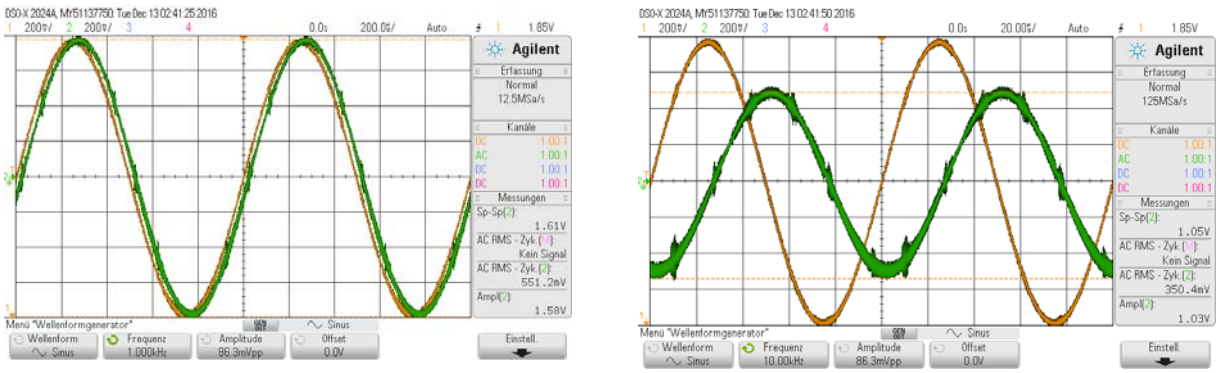
**Note:** In the Electronics Lab of *OTH Regensburg*, use BNC plug *Gen Out* of *DSO-X* oscilloscope as waveform generator. Button *WaveGen* pops up the respective screen menu.

- Fig. 2.1(c) should sketch the jumpers to obtain a 1st order analog system.
- Realize these settings with jumpers on the DSM board.
- We want to feed a 2 $V_{pp}$ sinusoidal Signal into the *DSM* board's $U_{in}$. To do so, we feed ca. 200 $mV_{pp}$ into the *ADA*-board's Pin A34 or TRS connector and jumper pins A35-A37. Adjust input voltage to pin A34 such, that you get the yellow input curves $U_{in}$ with 2 $V_{pp}$ at pin A39 as shown in Fig. 3.1 and the green output curves at pin E5. (Hint: Observation is easier if signals are in phase, i.e. if you set 2nd order system on the *DSM* board.)
- The cutoff frequency of DSM board's analog lowpass is ca. 10KHz. Measure the output signal at pin E5 or E6 for $f_{in}$ = 1KHz, 10KHz and 100KHz. What amplifications do you observe?

  1st order: 1 KHz: A = . . . . . dB, 10 KHz: A = . . . dB, 100 KHz: A = . . . . . dB,

- Modify a single Jumper such, that we have a 2nd order analog system. Feed $U_{in}$ as detailed above and measure amplifications at pin E5 or E6:

  2nd order: 1 KHz: A = . . . . . dB, 10 KHz: A = . . . . dB, 100 KHz: A = . . . . . dB,




(a) at 1 KHz at 2nd order system (b) at 10 KHz at 2nd order system

**Fig. 3.1:** Adjust *DSM* board input voltage (yellow) to fit to the screen, observe output (green)

## 3.2 Testing *DSM* Board as Submodule of *ADA* Board

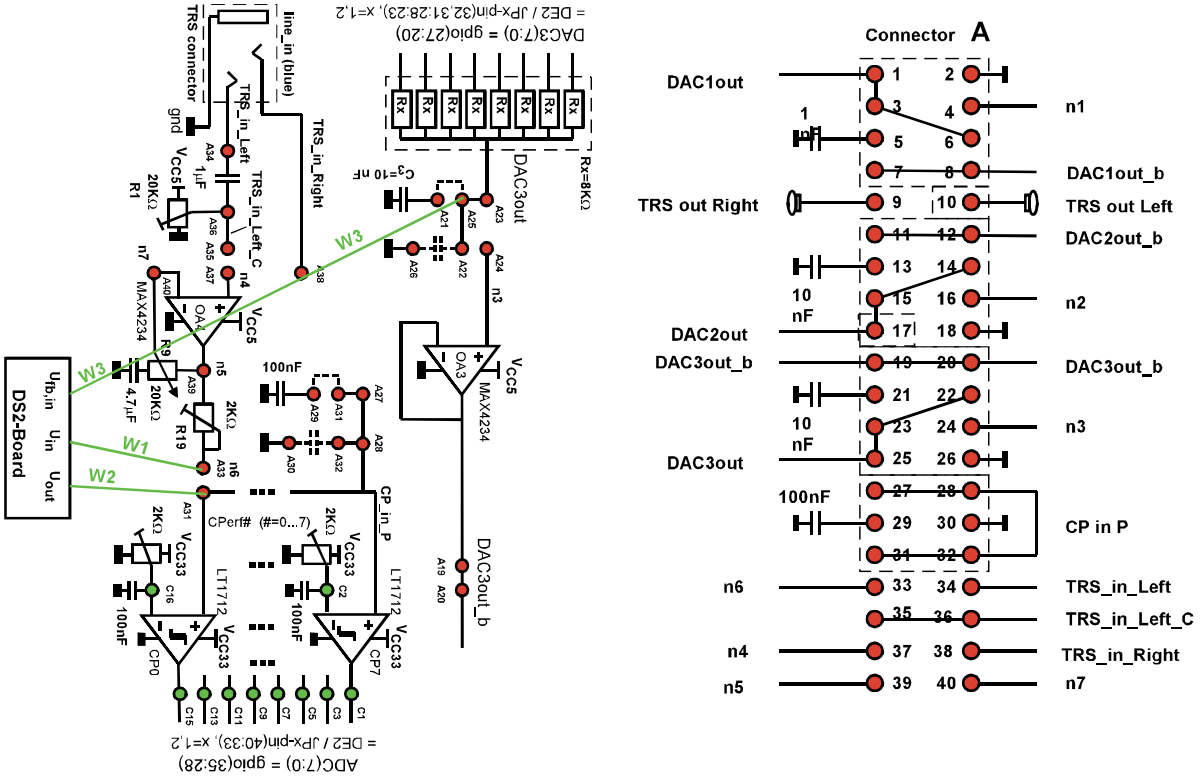



**Fig. 3.2:** Schematics and jumper situation

**We now make a similar measurement as in the previous chapter illustrated in Fig. 3.1, but we close the loop not on the *DSM* board but no the *ADA* board.**

- Remove jumper E3-E4 on *DSM* board.
- To connect the wire *W2* in Fig. 3.2 set jumper E1-E3 on the *DSM* board.
- To connect the wire *W3* in Fig. 3.2 set jumper E2-E4 on the *DSM* board.
- On the *ADA* board, set jumper A25-A27. Sketch jumper settings (pink colour) in Fig. 3.2 left and right. (With these jumper settings output amplifier $OA_1$ on the *DSM* board overrides the 1KΩ-output impedance of *DAC3*, as we cannot disconnect *DAC3*.

Fig. 2.1(d) should sketch the desired configuration with jumpers to obtain an a 2nd order system on *DSM* board with $DSM/U_{out} = ADA/A27$ and $DSM/U_{fb,in} = ADA/A25$.

Experimental verification: We should get the same results as in the subchapter before:

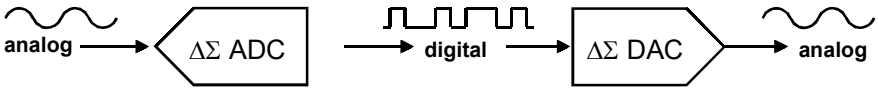
- 1st order: 1 KHz: A = . . . . . dB, 10 KHz: A = . . . . . dB, 100 KHz: A = . . . . . dB,
- 2nd order: 1 KHz: A = . . . . . dB, 10 KHz: A = . . . . . dB, 100 KHz: A = . . . . . dB.

The ADA conversion system is complete when we remove the (in Fig. 3.2 pink) jumper A25-A27.

# 4 *ADAC*: A Delta-Sigma A/D/A Conversion System

## 4.1 *ADAC* System Overview

**(a)** Top level system Overview.




**(b):** The complete system: A ΔΣ-ADC feeds *DAC1* and *DAC2*

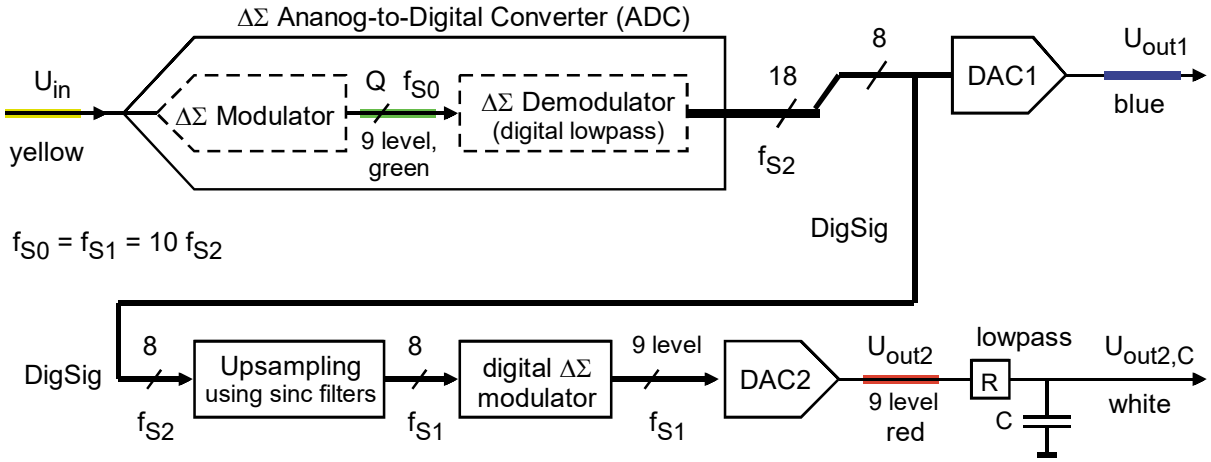



**Fig 4.1:** A/D – D/A Conversion System overview.

The complete system under consideration is illustrated in Fig. 4.1(b) and 4.2. Input $U_{in}$ (yellow) is fed to a ΔΣ ADC consisting of modulator and demodulator connected by quantized signal Q (green). It delivers signal *DigSig*, which is translated directly to $U_{out1}$ (blue) by the 256-level R2R-DAC named *DAC1*, and indirectly to $U_{out2}$ (red) by the 9-level DAC named *DAC2*, and after smoothing to $U_{out2,C}$ (white).

## 4.2 Simulation Using ModelSim

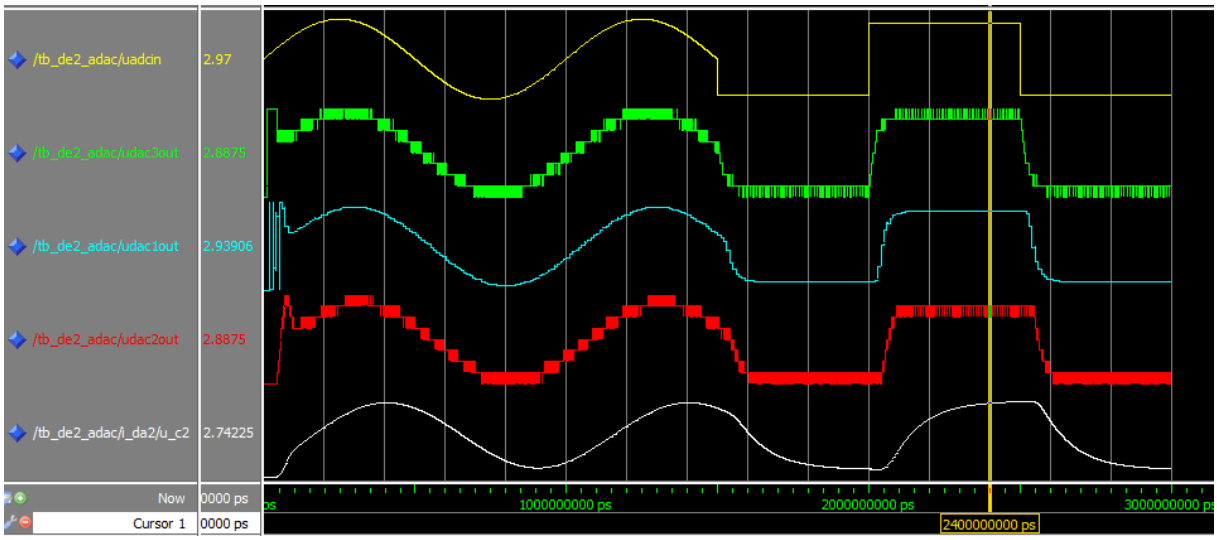



**Fig. 4.2:** CompleteADA conversion system simulation with *ModelSim*.

For the following simulation and synthesis load and unzip file *adac.zip* [16], [17]. You get directory

```
<treename>/adac/ModelSim/testbenches/tb_de1soc_adac/
```

where you find the files

```
tb_de1soc_adac.vhd  % VHDL testbench
wave.do             % controls the wave window (graphics output) of ModelSim
work.do             % compiles all files for the students
```

Start *ModelSim* simulator, use **`Files → Change Directory`** to navigate into directory **`tb_de1soc_adac`**. After executing *Files → Quit* here you can navigate more easily to this directory using *Files → Recent Directories*.

With *ModeSim* working directory set to *tb_de1soc_adac* type into the *transcript* window

```
> do work.do
```

The **`do`**-command processes commands within file *work.do*, which uses the graphics command file *wave.do*. You should get the result shown in Fig. 4.2.

**Questions:** ΔΣ converters are said to be accurate but slow, slow because of the big modulator's delay. This statement is both true and wrong. Argue with Fig. 4.2!

. . . . . .

. . . . . .

# 4.3 Theory (=Green Text) of ΔΣ A/D and D/A Conversion

## 4.3.1 Goals of ΔΣ Modulation

A delta-sigma (ΔΣ) system consists of a modulator / demodulator combination.

**Modulator settings:** The modulator translates signal resolution to speed (expressed as oversampling). In this example we will use a *NoL* (number of) levels quantizer, corresponding to an effective number of bits of $ENOB = ld(L) \Leftrightarrow 2^{ENOB}=NoL$. We can set *NoL* by toggle switches $sw(9:8)$="00", "01","10", "11" to $NoL = 2^{sw(17:16)}+1 = 2, 3, 5, 9$ corresponding to $ENOB = ld(9) = 1, 1.6, 2.3, 4.2$. The logarithm dualis can be computed as $ld(x)=ln(x)/ln(2)$ or with *Matlab* as *log2*(x).

Demodulator: The demodulating lowpass re-translates speed to resolution. For DC applications simple averaging is enough. If we want to average busy signals a lowpass delivers constant amplification for all baseband frequencies.

**ΔΣ D/A Converter:** The most important modulator type particularly for DACs is the 1-bit resolution type (i.e. 2 levels only). This is because loads can be driven with theoretically 100% energy efficiency using a simple switch: Few power is lost in the switches: Either there is no current through or no voltage across them. A speaker or headphone can be driven with a ΔΣ modulated bitstream with near 100% energy efficiency. The demodulating lowpasses in these example is the physical mass of the speaker.

**A/D Converter:** A further important goal of oversampling ADCs is the avoidance of analog anti-aliasing filters. The unavoidable lowpass filtering is shifted into the digital domain. When oversampling is employed to avoid anti-aliasing filtering, then ΔΣ modulation might make sense to yield more accuracy.

## 4.3.2 Principle of ΔΣ Modulation

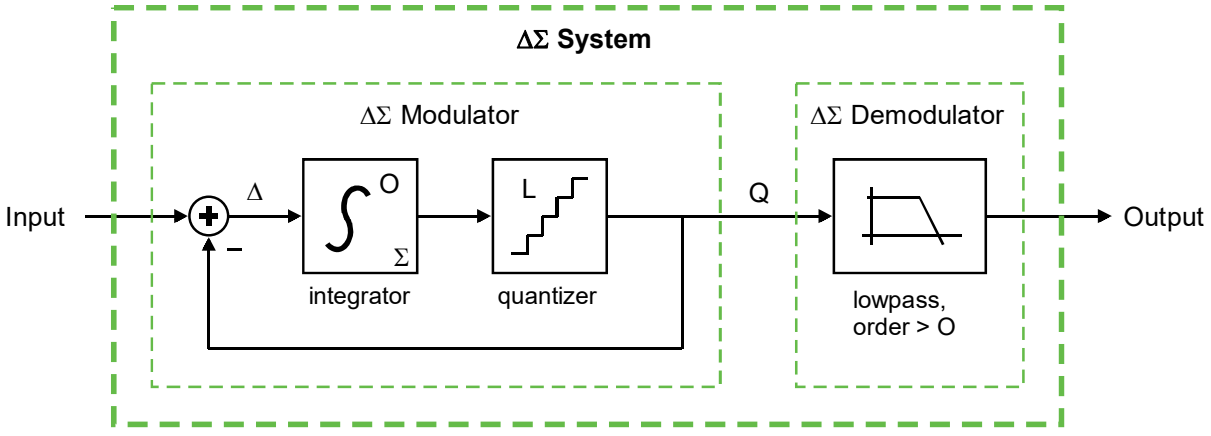



**Fig 4.3.2:** Principle of ΔΣ Modulation: An $O^{th}$ order integrator and a quantizer within a loop

Fig. 4.3.2 illustrates the basic principle of ΔΣ modulation: A loop of an $O^{th}$ order integrator followed by a *L* - level quantizer. On the one hand, the quantizer makes the signal less accurate, on the other hand, the accuracy is preserved as average of the faster clocked loop.

Consequently, demodulation is forming the average of the quantized data stream Q. This averaging within base-band frequencies is performed by the lowpass.

As illustrated in Fig. 4.3.2 the "delta" (Δ) is the difference between input signal and quantized output signal Q. The integrator, symbolized by the Greek letter "sigma" (Σ), sums all deviations between input signal and Q while the loop seeks to minimize this sum.

## 4.3.3 Using ΔΣ Modulation for D/A Conversion

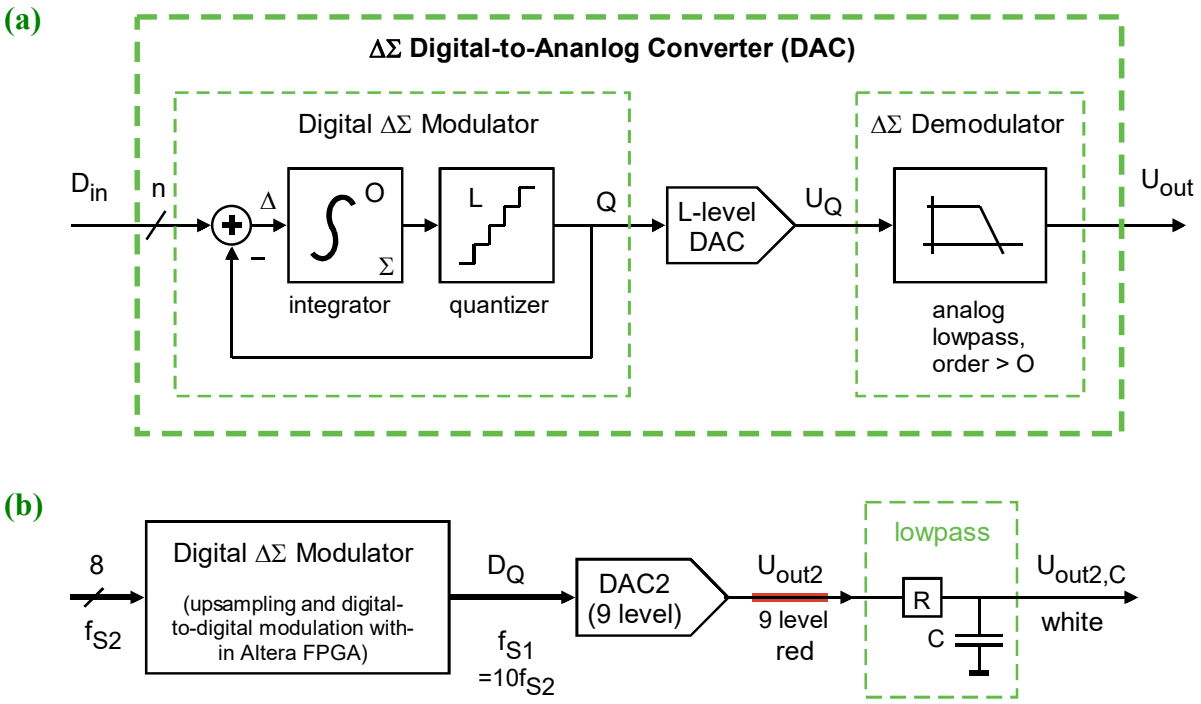



**Fig 4.3.3: (a)** Using ΔΣ Modulation for D/A Conversion. **(b)** Realization in this lab.

Fig 4.3.3 illustrates the principle of using an ΔΣ modulator for D/A conversion. We have a completely digital modulator with digital input $D_{in}$ and digital output $D_Q$. The quantizer can be realized by simply omitting some of the lower significant bits, in the extreme situation $D_Q$ is one bit only. To compensate for the loss of lower significant bits we have an increased clock output speed $f_{S1} = K_1 \cdot f_{S2}$. The DAC is required to translate the output signal from digital to analog domain.

In this lab we have $K_1 = 10$ (consequently $f_{S1} = 10 \cdot f_{S2}$).

In this lab *DAC2* of the ADA board is fed with an 8-bit / 9-level thermometric code. Its possible states are "00000000" , "00000001", "00000011", "00000111", "00001111", "00011111", "00111111", "01111111", "11111111". The output levels are shown as red curves in Figs. 4.2 and 4.5.

In this lab the Digital-to-digital (D/D) modulator has a 1st order integrator. Consequently, the demodulator would require at least 2nd order lowpass to exploit possible signal-to-noise ratio (SNR). Here we yielded optically acceptable results with a simulated 1st order RC lowpass, shown as white curve in Fig. 4.2.

## 4.4 Using ΔΣ Modulation for A/D Conversion

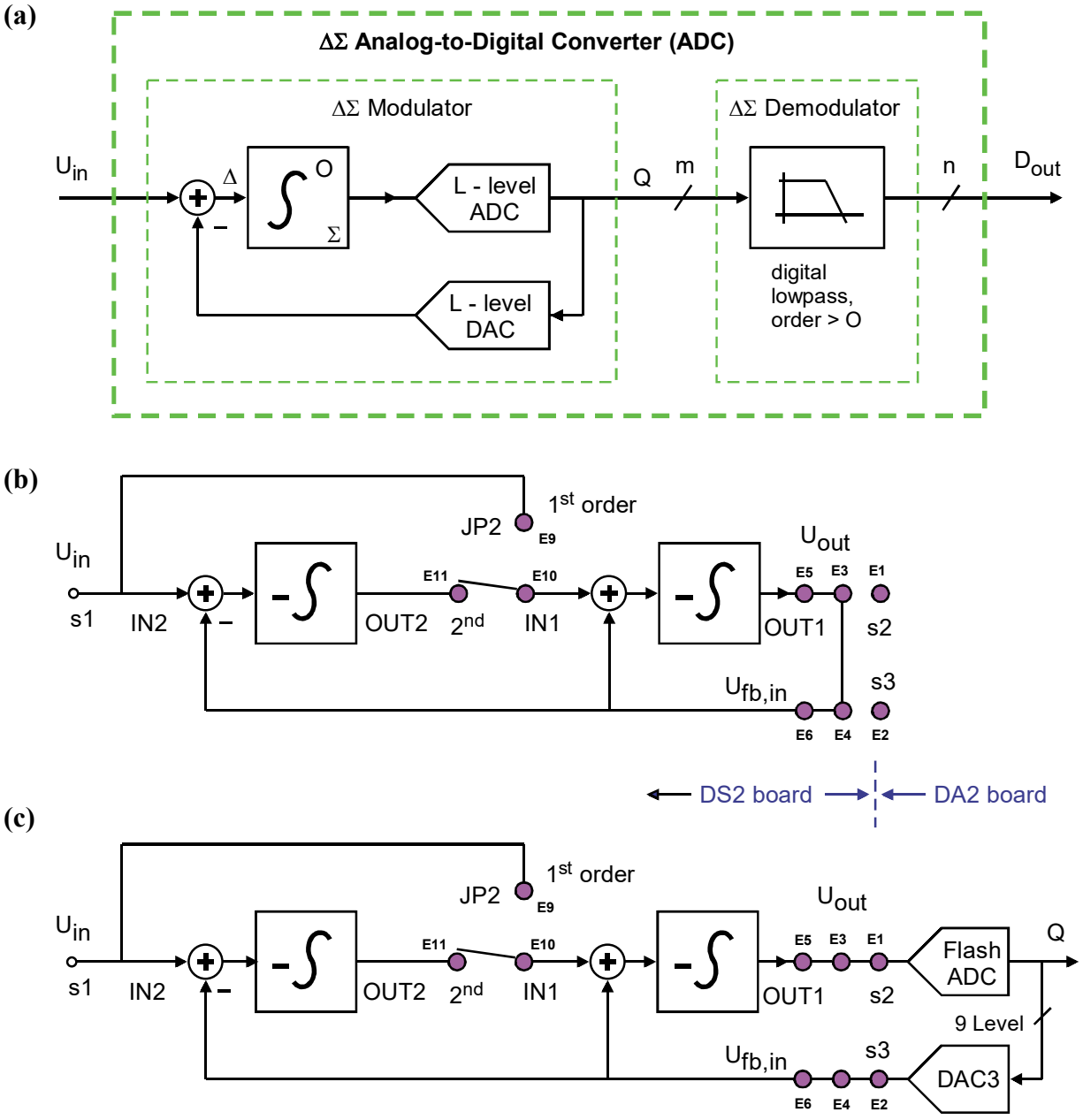



**Fig 4.4.1: (a)** Principle of using ΔΣ Modulation for A/D Conversion. **(b)** Analog integrator loop of DSM board. **(c)** Incorporation of ADC and DAC makes analog loop to ΔΣ modulator.

To build a ΔΣ modulator for A/D conversion we use an analog integrator of $O^{th}$ order and an ADC as quantizer that translates $U_{out}$ of the analog integrator to the quantized digital output $Q$ of the modulator. As the difference Δ has to be computed in the analog domain, a DAC is required to re-translate the digital output Q to the analog feedback signal $U_{fb,in}$.

In this lab we use according to Figs. 4.4.1 (a) and (b) a jumper on pins E9, E10, E11 to switch the order $O$ of the integrator and consequently the order of the modulator between $O$=1 and $O$=2.

Fig. 4.4.1(b) illustrates the integrator system operating as analog lowpass of 1st or 2nd order.

In Fig. 4.4.1(c) we make it a ΔΣ modulator by incorporation of the Flash-ADC (on *ADA* board) as quantizer into the loop. This requires to re-translate the digital signal into an analog signal, which is done by *DAC3* in the feedback branch.

**To do and to check:**

- *DSM* board: Input voltage $U_{fb,in}$ is connected to ADA-baord's DAC3 output by wire *W3* in Fig. 4.4.2. Do: Confirm that jumper E2-E4 on DSM board is set acc. to Fig. 3.1(d).
- *DSM* board: Output voltage $U_{out}$ is connected to ADA-baord's Flash-ADC input *CP_in_p* by wire *W2* in Fig. 3.2. Confirm jumper E1-E3 on DSM is set acc. to Fig. 3.1(d).
- *ADA* board: Do: Complete input signal path: Set a jumper on connector A to connect *TRS_in_left_C* to *n4*. Plot the jumper in both left and right side of Fig. 3.2.
- *ADA* board: Check: Remind that the AC input signal must proceed from *n4* to *n5* with amplification of 10. Amplification can be adjusted with poti *R9*.
- *ADA* board: Check: Remind that a sinusoidal signal with peak-to-peak amplitude of 280 mV on *n4* must deliver a sinusoidal signal of 3V peak-to-peak on *n5*. Center waveforms to $V_{CC}/2$=1.65V with poti *R1* if necessary.

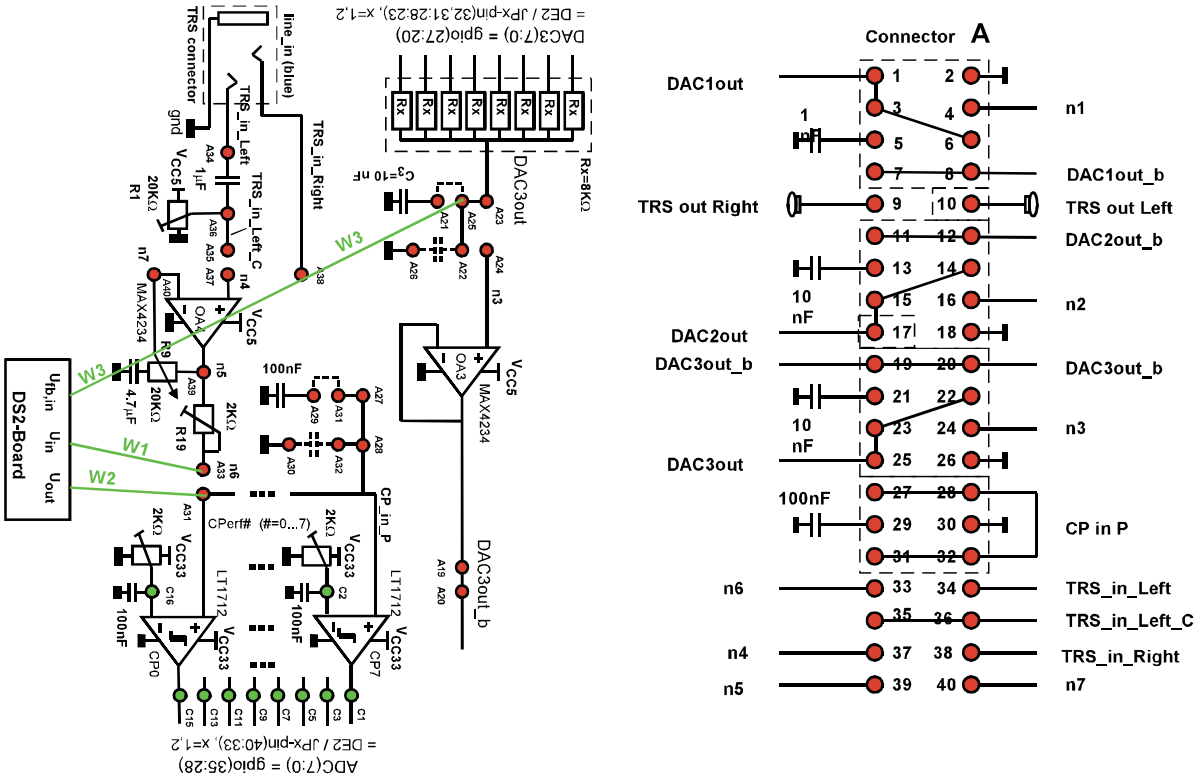



**Fig 4.4.2:** Schematics and jumper situation

## 4.5 Hardware Test Using *Quartus II 8.1*

*DE1-SoC* board: Make sure that only jumper A35-A37 is set on connector A
DSM board: Connected by the jumpers E1-E3 and E2-E4. Set ΣΔ order, e.g. jumper E10-E11.
The green wires in Fig. 4.4.2 are automatically connected by plugging in the *DSM* board.
Download file *ci_de1soc.zip* frothe author's homepage [15] and unzip it.

PC: Start *Quartus II Rev. 18* or later. You have two possibilities to accomplish that:
(i) In directory *ci_de1soc_adac* double-click on file *ci_de1soc_adac.qpf*.
(ii) Start *Quartus II*, navigate **`File → Open Project → ... → ci_de1soc_adac.qpf`**.

- Make sure that **`Hardware Setup`** is **`USB Blaster`**.
- Select: **`Tool → Programmer → Add File... → ci_de1soc_adac_demo.sof`**
- Make sure that only flag **`Program/Configure`** is set in the programmer, then click **`Start`**.
- Let's now work get the oscillogram shown in Fig. 4.5. The channels were selected to match the colors of the *ModelSim* simulation shown in Fig. 4.2.

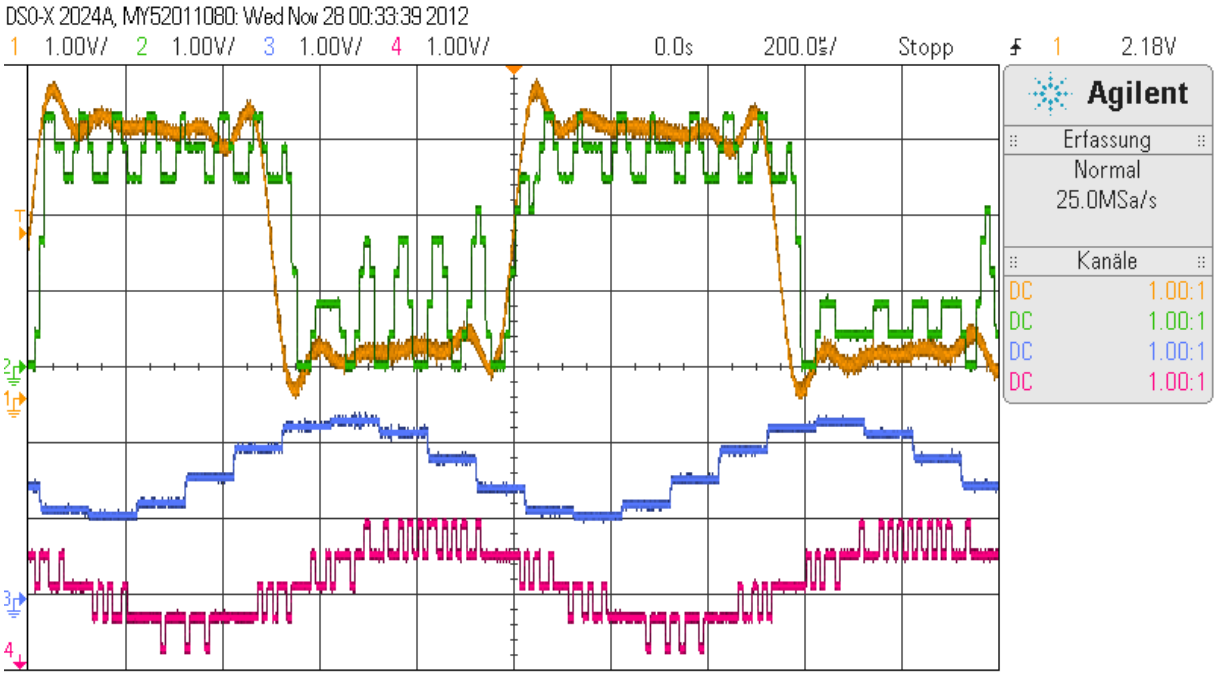



**Fig 4.5:** Oscillogram: $U_{in}$ (yellow) and $Q$=$U_{fb,in}$ (green) of DSM board, $U_{out1}$ (blue), $U_{out2}$ (red).

The curves seen in the oscillogram in Fig. 4.5 are:

- CH1: Top yellow smooth curve: $U_{in}$ to DSM board, measured at pin **`ADA/A39 = DSM/s1`**, the rectangular curve got oscillations from ADA-board OpAmp $OA_4$.
- CH2: Top green 9-level curve = ΔΣ modulator's output Q from flash-ADC, measured as output of **`DAC3 (pin ADA/A25)`** = input $U_{fb,in}$ of DSM board.
- CH3: Blue: Output of R2R-DAC named *DAC1* **`(pin ADA/A1)`**, 256-level, being the visualization of ΔΣ -ADC output *DigSig*. The time delay of 65 cycles of sampling clock $f_{S2}$ is due to the ΔΣ -demodulator (=lowpass).

- CH4: Lower red 9-level curve: Output $U_{out2}$ of ΔΣ DAC named *DAC2* **`(pin ADA/A17)`**. Input was the ADC's output *DigSig* (blue). The time delay compared to CH3 is due to the interpolating sinc$^4$ filter.

**Check for keys:**

- Set Switches *sw*(17:0)="11 0000 0000 0000 0101" on *DE1-SoC* Board.
- Test *reset* signal *key*(0), global *enable* signal *key*(1) while *sw*(3)='0'.
- Test manual clock *key*(2) when *sw*(3)='1'. (Probably not much to see.)

**Check for switches:**

- Start with Switches *sw*(17:0)="11 0000 0101" on *DE1-SOC* Board.
- ***sw(2:0)*="101":** Play with switches *sw(2:0)*. They set the sampling frequency of the system according to $f_{S0}=10^{sw(2:0)}$. The system will use $f_{S0}=f_{S1}=10{\cdot}f_{S2}$. A good setting is *sw(2:0)* = $101_2 = 5_{10}$. Then $f_{S0}=f_{S1}=10^5$Hz=100KHz and $f_{S2}$=10KHz allowing to sample waveforms up to $f_{S2}/2$ = 5KHz. Lower clock rates are problematic to hear sound, and higher sampling rates cause problems with parasitic capacitances.
- ***sw(9:8)*="11":** Switches *sw(9:8)* set the number of levels of the Flash-ADC to $NoL = 2^{sw(17:16)}+1$. Consequently, *sw(9:8)*="11" → *NoL*=9, "10" → *NoL*=5, "01" → *NoL*=3, "00" → *NoL*=2. (While the Flash-ADC always delivers 9 levels, *NoL* is reduced by logic in the VHDL code. In Figs. 4.2 and 4.5 we used *NoL* =9 levels.
- Set all the other toggle switches to zero: ***sw(7:3)* ="00000"**.

**Check for *ADA* board:**

- Feed a sinusoidal 1KHz signal with amplitude of 200mV$_{pp}$ to *TRS_in_Left* (pin *A34*). You may use the sound card of your computer or a waveform generator. This signal should appear with a DC offset of $V_{CC}/2$=1.65V at pin *A34* (=*n4*). Offset correction can be done with potentiometer *R1*.
- CH1: top, yellow: Signal $U_{in}$ of Fig. 4.1: Then switch your input waveform from a sinusoidal to a rectangular, still having 2V$_{pp}$. Now V(*n5*) should deliver the yellow curve shown in Fig. 4.5. It must be centered around $V_{CC}/2$=1.65V, range 0.65V - 2.65V. Adjust amplification of factor 10 with poti *R9* if necessary.
- CH2: top, green: Quantized signal *Q* of Fig. 4.1: Connect channel 2 of your oscilloscope to pin **`A25`** of **`ADA`**. You should get the top green 9-level curve measured at **`pin ADA/`**. This is the ΔΣ modulator's output *Q* fed to **`DAC3`** delivering = $U_{fb,in}$ of DSM board. Observe impact of switches *sw(17:16)* on the number of levels of this waveform.
- CH3: mean, blue: Signal *DigSig* of Fig. 4.1: Connect channel 3 of your oscilloscope to pin **`A1`** of **`ADA`**. This is the output of the R2R-DAC named *DAC1*, featuring 256-levels, which visualizes the ΔΣ –ADC's output *DigSig*. The time delay of 65 cycles of sampling clock $f_{S2}$ is due to the ΔΣ –demodulating low-pass. Note that sampling frequency is $f_{s2} = f_{s0}/10$.
- CH4: bottom, red: Signal $U_{out2}$ of Fig. 4.1: Connect channel 4 of your oscilloscope to pin **`A17`** of **`ADA`**. This is $U_{DAC2out}$, the output of *DAC2*. The input signal of *DAC2* was obtained from the blue signal *DigSig* by speeding its sampling rate up by a factor 10 to obtain $f_{s1}=10{\cdot}f_{s2}$. With this higher sampling rate it was ΔΣ modulated and fed to *DAC2*. The time delay compared to CH3 is due to the interpolating sinc$^4$ filter, necessary after up-sampling.

**Exercises:**

Observe Fig. 4.5. Is the ΔΣ modulator fast (green following yellow curve)? **...yes....**

Observe $U_{out1}$ (*DAC1out*, blue). Who swallowed the rectangular edges of $U_{in}$ (yellow)?

. . . . . .

Observe from Fig. 4.5 if the modulator is operated at 1st or 2nd order. To do so keep in mind, that a 1st order modulator will never do jumps over 2 Δ's without significant jumps of the input signal.

. . . . . .

. . . . . .

# 5 Conclusions

This communication demonstrated how the DSM board can e used to assemble a ΔΣ ADC with the author's *ADA* board and *Terasic's DE1-SoC* board.

# 6 References


[1] Available: https://www.terasic.com.tw.
[2] Available: https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&No=836
[3] Available: http://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&No=886
[4] Available: https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=165&No=836&PartNo=4
[5] CD ROM *DE1-SoC CD-ROM (rev.F Board) Version 5.1.2 of 2910-01-28* from [4]
[6] *DE1*-SoC User Manual, Ref. F, taken from [5]
[7] *DE1*-SoC Schematic, Ref. F, taken from [5]
[8] K:\SB\Sources\EDA\Terasic\Hardware\
[9] Available: https://en.wikipedia.org/wiki/Intel_Quartus_Prime
[10] Available: https://en.wikipedia.org/wiki/ModelSim
[11] Av. https://www.intel.com/content/www/us/en/programmable/downloads/download-center.html
[12] Available: https://edg.uchicago.edu/~tang/VHDLref.pdf
[13] Available: https://www.mimuw.edu.pl/~marpe/pul/card_vhdl.pdf
[14] Available: https://www.mimuw.edu.pl/~marpe/pul/card_1164.pdf
[15] Available: https://hps.hs-regensburg.de/scm39115/
[16] MAX4230 - MAX4324 data sheet, Maxm Integerated, Available: https://datasheets.maximintegrated.com/en/ds/MAX4230-MAX4234.pdf
[17] LT1711 / 1712 Rail-to-Rail Comparators, Analog Devices (Linear Technlogy) Available: https://www.analog.com/media/en/technical-documentation/data-sheets/171112f.pdf